

Canon EOS Digital SDK

RC-SDK8.2 API Programming Reference

Part III : Development Edition

Revision 1.10
11/19/2004

The information contained in this document is subject to change without notice. Canon Inc. makes no warranty of any kind with regard to this material, either express or implied, except as provided herein, including without limitation thereof, warranties as to marketability, for a particular purpose or use, or against infringement of any patent. Canon Inc. shall not be liable for any direct, incidental, or consequential damages of any nature, or losses or expenses resulting from the use of this material.

This document contains proprietary information, which is protected by copyright. All rights are reserved. No part of this document may be photocopied, reproduced, or translated into another language without prior written consent of Canon Inc.

Considerable effort has been made to ensure that this manual is free of inaccuracies and omissions. However, as we are constantly improving our products, some of the data contained herein may not exactly reflect the current model of the particular product with which this manual has been included.

Copyright © 2004 Canon Inc.

Canon, and EOS are registered trademarks of Canon Inc. in Japan and/or other countries.

Windows is a registered trademark of Microsoft Corporation in the United States and/or other countries.

Macintosh is a registered trademark of Apple Computer, Inc. in the United States and/or other countries.

All other trademarks in this documentation are registered trademarks and trademarks of their respective owners.

Revision History

Revision No.	Version	Date	Revised page(s)	Reason and content of revision
-	1.00	5/12/2004		First revision release
	1.10	11/19/2004		Description about EOS 20D and EOS-1Ds MarkII was added.

Table of Contents

1. INTRODUCTION	1
1.1 RAW Data Formats	1
1.2 Setting Development parameter	1
2. OVERVIEW	1
3. DATA TYPE DEFINITION	2
3.1 Basic	2
3.2 Output file format	2
4. DESIGN OF THIS DOCUMENT	3
4.1 Function Name	3
5. RAW OBJECT HANDLING	4
5.1 RCCreateRAWObject	4
5.2 RCDisposeRAWObject	4
6. AVAILABLE DEVELOPMENT PARAMETER ITEM	5
6.1 RCIsSupportParamItem	5
6.2 RCGetSupportParamItemHead	6
6.3 RCGetSupportParamItemNext	6
7. AVAILABLE DEVELOPMENT PARAMETER VALUE	8
7.1 RCIsSupportParamValue	8
7.2 RCGetSupportParamValueHead	8
7.3 RCGetSupportParamValueNext	9
8. DEVELOPMENT PARAMETER SETTINGS	10
8.1 RCGetParamValue	10
8.2 RCSetParamValue	12
8.3 RCSetAllParamValueToAtCapture	12
8.4 RCCopyClickWBCoefs	13
8.5 RCReflectParamValues	13
9. RAW DEVELOPMENT	15
9.1 RCDevelop	15
10. GET DEVELOPMENT PARAMETER AND UPDATE IMAGE	17
10.1 RCWriteDevParamToRAW	17
10.2 RCCopyRawObjPrmToStgMedium	17
10.3 RCCopyStgMediumPrmToRawObj	18
10.4 RCWriteParamValueFile	18
10.5 RCReadParamValueFile	19
11. MEMORY UTILITY	20
11.1 RCUnFocusRAWObject	20

Revision History/Date	Corrections	Reviser	Remarks

1. Introduction

This RC-SDK 8.0 API Reference, Part III Development Edition describes the functions used to develop RAW data and convert it to JPEG and other formats. RAW is one type of image data format of Canon digital camera, and no image processing is performed on it within the camera. It is data recorded from the captured pixels as is. RAW image data formats include cdFORMAT_RAW and cdFORMAT_RAW_SPATIAL data. You can discover the data format by examining the basic image data properties.

1.1 RAW Data Formats

The format of the RAW data depends on the model of the camera used to take the picture. Some of the development parameter types and values that can be set may differ depending on the camera model. Therefore, to set development parameters or convert the RAW data to develop it, you must first retrieve and evaluate the format of the RAW data.

There are different variations on RAW file. RAW file format is distinguished by the extension of the file name. The image files that have an extension of CRW, TIF and CR2 are just the RAW image data.

Table 1-1 Extension and Camera Model

Extension	Camera Model
CRW	D30, D60, 10D, Kiss Digital, D300, Digital Rebel
TIFF	1D, 1Ds
CR2	1D Mark II, 20D, 1Ds Mark II

1.2 Setting Development parameter

When you develop RAW data, you can change the development parameters, including the application of a filter or the white balance setting. The functions given here are for setting development parameters. However, note that non-supported functions and values that can be set may differ depending upon the RAW development faculty value.

To set development parameters and develop the data, execute these functions, make the settings in the RC-SDK, and then execute the RCDevelop() function described later that develops the RAW data.

Revision History/Date	Corrections	Reviser	Remarks

2. Overview

This chapter describes overview of the functions used for RAW development process. There are many functions related to RAW development process, so they are divided into the following groups for purposes of description

RAW object create/dispose Functions

- RCCreateRAWObject
- RCDisposeRAWObject

Get available development parameter item

- RCIsSupportParamItem
- RCGetSupportParamItemHead
- RCGetSupportParamItemNext

Get available development value

- RCIsSupportParamValue
- RCSupportParamValueHead
- RCSupportParamValueNext

Get/Set current development parameter

- RCGetParamValue
- RCSetParamValue
- RCSetAllParamValueToAtCapture
- RCCopyClickWBCoefs

Development process

- RCDevelop

Get development parameter and update image data

- RCWriteDevParamToRaw
- RCCopyRawObjPrmToStgMedium
- RCCopyStgMediumPrmToRawObj

Memory utility function

- RCUnFocusRAWObject

Revision History/Date		Corrections	Reviser	Remarks

3. Data Type Definition

3.1 Basic

The definition of the parameter related to RAW development is described to rcType.h. The old definition value described in cdType.h cannot be used in RCSDK.

The cdStgMedium structure is a generalized memory handle used for data transfer operations by the RCSDK interface.

3.2 Output file format

The header file rcType.h has data type definition used by only RCDevelop function.

```

Typedef cdUInt8 cdDevTgtFormat

#define cdTGT_FORMAT_JPEG                2
#define cdTGT_FORMAT_PIXMAP              4
#define cdTGT_FORMAT_JPEG_EXIF           5
#define cdTGT_FORMAT_PIXMAP_16           7
#define cdTGT_FORMAT_TIFF_EXIF           8
#define cdTGT_FORMAT_TIFF                 9
#define cdTGT_FORAMT_BMP                  13
#define cdTGT_FORMAT_BMP_TOPDOWN         14

```

Revision History/Date	Corrections	Reviser	Remarks

4. Design of this document

This API reference has the following format in each RC-SDK API.

4.1 Function Name

Description

Describes the detail of API.

Syntax

Adopts the expression of C language in function definition

[Type of return value] [Calling rules] [Function name (parameter type parameter name ...)]

Parameters

Parameter and parameter type to call the API

Return Values

Specify the value that API terminates and returns.

See Also

Referential section of the API (Option)

Note

Handling instruction (Option)

Revision History/Date	Corrections	Reviser	Remarks

5. Raw object handling

This section introduces the functions used for basic operations, such as initiating RAW development.

5.1 RCCreateRAWObject

Description This function creates a RAW object that is associated with the specified cdStgMedium structure. The client needs the RAW object when using each development API. The client should use RCDiscardRAWObject to dispose a Raw Object.

Syntax **cdCAPI RCCreateRAWObject(**
cdStgMedium* pStgMedium, cdHRAW * phRAW)

Parameters

pStgMedium cdStgMedium*; specify the pointer to cdStgMedium structure.

phRAW cdHRAW*; specify the pointer to the RAW handle that this function returns RAW object handle into the address specified by this parameter.

Returns cdError: cdOK if successful; otherwise, the return value is one of the bellow error list.

Error Type	Comments
cdNOT_SUPPORTED	Not Support
cdINVALID_PARAMETER	Parameter error
cdMEM_ALLOC_FAILED	Memory allocation error

5.2 RCDiscardRAWObject

Description This function disposes the specified RAW object.

Syntax **cdCAPI RCDiscardRAWObject (cdHRAW hRAW)**

Parameters

hRAW cdHRAW; specify the handle of RAW object

Returns cdError: cdOK if successful; otherwise, the return value is one of the bellow error list.

Error Type	Comments
cdINVALID_HANDLE	Invalid handle
cdMEM_ALLOC_FAILED	Memory allocation error
cdNOT_SUPPORTED	Not Support

Revision History/Date	Corrections	Reviser	Remarks

6. Available development parameter item

This section describes the functions used to get the available development parameter items.

6.1 RCIsSupportParamItem

Description This function determines whether the RAW object can set the specified development parameter item.

Syntax `cdCAPI RCIsSupportParamItem(cdHRAW hRAW,
cdDevT_ParamItemID prmID, cdDevT_Availability* pAvailable)`

Parameter

hRAW cdHRAW; specify the handle of the RAW object

prmID cdDevT_ParamItemID; specify the ID of development parameter item.

Value	Meaning
kPrmID_16bitLinear	16bit linear
kPrmID_DgtlExpCmp	Digital exposure compensasion
kPrmID_LightSrc	Light source of white balance
kPrmID_PosClickWB	Click white balance coordinates
kPrmID_ColorTemp	Color temperature
kPrmID_MgGr	White Balance shift Magenta-Green
kPrmID_AmBl	White balance shift Amber-Blue
kPrmID_WBCoeffs	White Balance Coeffs
kPrmID_Contrast	Contrast
kPrmID_ClrTone	Hue
kPrmID_Saturation	Saturation
kPrmID_Sharpness	Sharpness
kPrmID_SharpnessTarget	Target applied to sharpness
kPrmID_ColorSpace	Color space
kPrmID_ColorMatrix	Color matrix(for 1D/1Ds/1D Mark II/1Ds Mark II)
kPrmID_ColorEffect	Color Effect
kPrmID_4ptRGBToneCurve	RGB Tone curve: 4Points(for 1D/1Ds)
kPrmID_12ptRGBToneCurve	RGB Tone curve: 12Points(for 1D Mark II/1Ds Mark II)

pAvailable cdDevT_Availability* ; pointer to cdDevT_Availability variable that this function returns status into the address specified this parameter, if this prmID parameter item is available.

Value	Meaning
kDev_NotSupported	Not supported in this camera model
kDev_AvailableToSet	Available

Returns cdError: cdOK if successful; otherwise, the return value is one of the bellow error list.

Error Type	Comments
cdINVALID_HANDLE	Invalid handle

Revision History/Date	Corrections	Reviser	Remarks

cdINVALID_PARAMETER	Parameter error
cdMEM_ALLOC_FAILED	Memory allocation error
cdNOT_SUPPORTED	Not Support

6.2 RCGetSupportParamItemHead

Description This function gets the settable development parameter items to the specified RAW object. To get the follow-on development parameter items after retrieving parameter item by calling RCGetSupportParamItemHead() , call RCGetSupportParamItemNext().

Syntax **cdCAPI RCGetSupportParamItemHead (cdHRAW hRAW, cdDevT_ParamItemID* pPrmID)**

Parameter

hRAW cdHRAW; specify the handle of the RAW object

pPrmID cdDevT_ParamItemID*; specify the pointer to ID of development parameter item.

Returns cdError: cdOK if successful; otherwise, the return value is one of the bellow error list.

Error Type	Comments
cdINVALID_HANDLE	Invalid handle
cdINVALID_PARAMETER	Parameter error
cdMEM_ALLOC_FAILED	Memory allocation error
cdNOT_SUPPORTED	Not Support

6.3 RCGetSupportParamItemNext

Description This function gets the follow-on settable development parameter items to the specified RAW object. In the case that the development parameter item is not exist any more, this function returns kPrmID_Invalid.

Syntax **cdCAPI RCGetSupportParamItemNext (cdHRAW hRAW, cdDevT_ParamItemID* pPrmID)**

Parameters

hRAW cdHRAW; specify the handle of the RAW object

pPrmID cdDevT_ParamItemID*; specify the pointer to ID of development parameter item.

Returns cdError: cdOK if successful; otherwise, the return value is one of the bellow error list.

Error Type	Comments
RCINVALID_PARAMETER	Invalid parameter entered.

Revision History/Date	Corrections	Reviser	Remarks

RCINVALID_HANDLE	hSource is invalid handle.
cdMEM_ALLOC_FAILED	Memory allocation error
cdNOT_SUPPORTED	Not Support

Revision History/Date	Corrections	Reviser	Remarks

7. Available development parameter value

This section provides the functions related to retrieve the development parameter values.

7.1 RCIsSupportParamValue

Description This function determines whether the RAW object can set the specified development parameter value.

Syntax

```
cdCAPI RCIsSupportParamValue( cdHRAW hRAW,
                             cdDevT_ParamItemID prmID,
                             cdDevT_RawPrmVals prmVal,
                             cdDevT_Availability* pAvailable )
```

Parameter

hRAW cdHRAW; specify the handle of the RAW object

prmID cdDevT_ParamItemID ; specify the ID of development parameter item.

prmVal cdDevT_RawPrmVals ; speicify patameter value to be checked

pAvailable cdDevT_Availability* ; pointer to cdDevT_Availablity variable that this function returns status into the address specified this parameter if this prmID parameter item is available.

Returns cdError: cdOK if successful; otherwise, the return value is one of the bellow error list.

Error Type	Comments
cdINVALID_HANDLE	Invalid handle
cdINVALID_PARAMETER	Parameter error
cdMEM_ALLOC_FAILED	Memory allocation error
cdNOT_SUPPORTED	Not Support

7.2 RCGetSupportParamValueHead

Description This function gets the settable development parameter values to the specified RAW object. To get the follow-on development parameter value after retrieving parameter value by calling RCGetSupportParamValueHead() , call RCGetSupportParamValueNext().

Syntax

```
cdCAPI RCGetSupportParamValueHead ( cdHRAW hRAW,
                                     cdDevT_ParamItemID PrmID, cdDevT_RawPrmVals* pPrmVal )
```

Parameter

hRAW cdHRAW; specify the handle of the RAW object

PrmID cdDevT_ParamItemID; specify the ID of development parameter item.

pPrmVal cdDevT_RawPrmVals*; specify the pointer to variable to be stored as development

Revision History/Date	Corrections	Reviser	Remarks

parameter value.

Returns cdError: cdOK if successful; otherwise, the return value is one of the bellow error list.

Error Type	Comments
cdINVALID_HANDLE	Invalid handle
cdINVALID_PARAMETER	Parameter error
cdMEM_ALLOC_FAILED	Memory allocation error
cdNOT_SUPPORTED	Not Support

7.3 RCGetSupportParamValueNext

Description This function gets the follow-on settable development parameter value to the specified RAW object. In the case that the development parameter value is not exist any more, this function returns kPrmVal_Invalid.

Syntax `cdCAPI RCGetSupportParamValueNext (cdHRAW hRAW,
cdDevT_ParamItemID PrmID, cdDevT_RawPrmVals* pPrmVal)`

Parameter

hRAW cdHRAW; specify the handle of the RAW object

PrmID cdDevT_ParamItemID; specify the ID of development parameter item.

pPrmVal cdDevT_RawPrmVals*; specify the pointer to variable to be stored as development parameter value.

Returns cdError: cdOK if successful; otherwise, the return value is one of the bellow error list.

Error Type	Comments
cdINVALID_HANDLE	Invalid handle
cdINVALID_PARAMETER	Parameter error
cdMEM_ALLOC_FAILED	Memory allocation error
cdNOT_SUPPORTED	Not Support

See Also RCGetSupportParamValueHead

Revision History/Date	Corrections	Reviser	Remarks

8. Development parameter Settings

This section introduces the functions related to set/get the development parameter.

8.1 RCGetParamValue

Description This function retrieves the specified development parameter value.

Syntax `cdCAPI RCGetParamValue(cdHRAW hRAW,
cdDevT_PrmKind prmKind,
cdDevT_ParamItemID prmID,
cdDevT_RawPrmVals* pGetValue)`

Parameter

hRAW cdHRAW; A handle for the RAW object.

prmKind cdDevT_PrmKind ; specify the kind of parameter value to be get.

Value	Meaning
kPrmKind_AtCapture	setting as shot
kPrmKind_CurrentSetting	current setting
kPrmKind_SupportMinValue	minimum value to be supported
kPrmKind_SupportMaxValue	maximum value to be supported
kPrmKind_StandardValue	standard value
kPrmKind_DevelopSetting	current develop setting

prmID cdDevT_ParamItemID; specify the ID of development parameter item.

pGetValue cdDevT_RawPrmVals* ;specify the pointer to variable to be stored as development parameter value.

This value changes with development parameter items.

In the case that prmID is kPrmID_4ptRGBToneCurve, parameter value is the pointer to cdDevT_4PointsRGB structure.

In the case that prmID is kPrmID_12ptRGBToneCurve, parameter values is the pointer to cdDevT_12PointsRGB structure.

Otherwise, parameter values is the pointer to type of cdUInt32

Returns cdError: cdOK if successful; otherwise, the return value is one of the bellow error list.

Error Type	Comments
cdINVALID_HANDLE	Invalid handle
cdINVALID_PARAMETER	Parameter error
cdMEM_ALLOC_FAILED	Memory allocation error
cdNOT_SUPPORTED	Not Support

See Also RCSetParamValue

Revision History/Date	Corrections	Reviser	Remarks

Revision History/Date		Corrections	Reviser	Remarks

8.2 RCSetParamValue

Description This function sets the specified development parameter value. Note that calling RCReflectParamValues() in order to making a setup reflect, before calling RCDevelop().

The client can get available parameter by calling RCGetSupportParamValueHead() and RCGetSupportParamValueNext(). Furthermore, the client can determine whether the parameter is available by using RCIsSupportParamValue().

Syntax

```
cdCAPI RCSetParamValue( cdHRAW hRAW,
                        cdDevT_ParamItemID prmID,
                        cdDevT_RawPrmVals prmVal )
```

Parameter

hRAW cdHRAW; A handle for the RAW object.

prmID cdDevT_ParamItemID ; specify the ID of development parameter item.

prmValue cdDevT_RawPrmVals ; specify the value of development parameter value

Returns cdError: cdOK if successful; otherwise, the return value is one of the bellow error list.

Error Type	Comments
cdINVALID_HANDLE	Invalid handle
cdINVALID_PARAMETER	Parameter error
cdMEM_ALLOC_FAILED	Memory allocation error
cdNOT_SUPPORTED	Not Support

See Also RCReflectParamValues

8.3 RCSetAllParamValueToAtCapture

Description This function resets the current development parameter values to shot setting.

Syntax

```
cdCAPI RCSetAllParamValueToAtCapture( cdHRAW hRAW )
```

Parameter

hRAW cdHRAW; A handle for the RAW object.

Returns cdError: cdOK if successful; otherwise, the return value is one of the bellow error list.

Error Type	Comments
cdINVALID_HANDLE	Invalid handle
cdMEM_ALLOC_FAILED	Memory allocation error
cdNOT_SUPPORTED	Not Support

Revision History/Date	Corrections	Reviser	Remarks

8.4 RCCopyClickWBCoefs

Description When the client set a Click White Balance Value by calling RCSetParamValue functoin, the RAW object holds it. The client should call this function to copy the curret parameter value to other RAW object.

It needs that two RAW objects are generated from the same camera model.

Syntax **cdCAPI RCECopyClickWBCoefs(cdHRAW hRAWSrc, cdHRAW hRAWTgt)**

Parameter
hRAWSrc cdHSource; A handle for source RAW object.

hRAWTgt cdHSource; A handle for target RAW object

Returns cdError: cdOK if successful; otherwise, the return value is one of the bellow error list.

Error Type	Comments
cdINVALID_HANDLE	Invalid handle
cdMEM_ALLOC_FAILED	Memory allocation error

8.5 RCReflectParamValues

Description The RAW object is maintaining the status of two parameters; the parameters for UI, and the parameters for development. This function reflects one status to another. The RAW object holds the status of the parameter for UI in the case of setting the parameter by calling RCSetParamValue(). To develop with these setting parameters, the client should call this function before calling RCDevelop().

Syntax **cdCAPI RCReflectParamValues(cdHRAW hRAW, cdDevT_ParamCopyDirection CopyDir)**

Parameter
hRAW cdHRAW; specify the handle of the RAW object

CopyDir cdDevT_ParamCopyDirection; direction flag of parameter reflecting

Value	Meaning
kPrmCpyUI2Dvlp	Reflect the parameters for UI to the parameters for development
kPrmCpyDvlp2UI	Reflect the parameters for development to the parameters for UI

Revision History/Date	Corrections	Reviser	Remarks

Returns

cdError: cdOK if successful; otherwise, the return value is one of the bellow error list.

Error Type	Comments
cdINVALID_HANDLE	Invalid handle
cdINVALID_PARAMETER	Parameter error
cdNOT_SUPPORTED	Not Support

Revision History/Date	Corrections	Reviser	Remarks

9. Raw Development

This chapter presents the function that executes development for RAW data.

9.1 RCDevelop

Description This function develops the specified RAW object with the current development parameters. To reflect the values to be set by RCSetParamValue(), call RCReflectParamValues().

Syntax

```
cdCAPI RCDevelop( cdHRAW
                  cdDevT_DevAction
                  cdDevTgtFormat
                  cdDevT_Dimension
                  cdStgMedium*
                  cdProgressCallbackFunction*
                  cdContext
                  cdProgressOption
                  hRAW
                  Action,
                  tgtFormat,
                  AreaAndRatio,
                  pStgMedium,
                  pCallbackFunc,
                  Context,
                  ProgressOption )
```

Parameter

hRAW cdHRAW; A handle for the RAW object.

Action cdDevT_DevAction ; specify the action of development.

Value	Meaning
kDevAct_FullDevelop	Full size development
kDevAct_PrevDevelop	Reduction size development
kDevAct_ThumbDevelop	Thumbnail size development

TgtFormat cdDevTgtFormat ; specify the format of development target.

Value	Meaning
cdTGT_FORMAT_PIXMAP	PIXMAP
cdTGT_FORMAT_PIXMAP_16	16bit Pixmap
cdTGT_FORMAT_BMP	Bitmap
cdTGT_FORMAT_BMP_TOPDOWN	TOP DOWN Bitmap

AreaAndRatio cdDevT_Dimension ; specify the pointer to cdDevT_Dimension structure. The client can specify the area and the reduction ratio into this structure.

pStgMedium cdStgMedium* ; specify the pointer to cdStgMedium structure.

pCallbackFunc cdProgressCallbackFunction ; specify the address of a callback function.

Context cdContext ; specify the context

ProgressOption cdProgressOption ; specify the option for progress callback.

cdDevT_Dimension Structure

Revision History/Date	Corrections	Reviser	Remarks

```
typedef struct{
    cdRect      DevelopArea;
    cdInt32     ReductNumerator;
    cdSize      DeveloppedSize;
} cdDevT_Dimension;
```

Member

DevelopArea The rectangle of the image area which is developed. The coordinate at the upper left of image is (0,0).

ReductNumerator specify the denominator of the reduction ratio as kReductDenominator.

DeveloppedSize the return value is the size of developed image.

Returns

cdError: cdOK if successful; otherwise, the return value is one of the bellow error list.

Error Type	Comments
cdINVALID_HANDLE	Invalid handle
cdINVALID_PARAMETER	Invalid parameter
cdMEM_ALLOC_FAILED	Memory allocation error
cdNOT_SUPPORTED	Not Support

Revision History/Date	Corrections	Reviser	Remarks

10. Get development parameter and Update image

This section provides the functions used to write the development parameter in RAW data and retrieved the development parameter from RAW data.

10.1 RCWriteDevParamToRAW

Description This function writes the development parameter values to be set in RAW object into the specified storage. The specified storage needs to associate with the same camera model as the storage of the RAW object.

Syntax `cdCAPI RCWriteDevParamToRAW(cdHRAW hRAW, cdStgMedium* pReflect)`

Parameter

hRAW cdHRAW; A handle for the RAW object.

pReflect cdStgMedium*; specify the pointer to cdStgMedium structure.

Returns cdError: cdOK if successful; otherwise, the return value is one of the bellow error list.

Error Type	Comments
cdINVALID_HANDLE	Invalid handle
cdINVALID_PARAMETER	Parameter error
cdMEM_ALLOC_FAILED	Memory allocation error
cdNOT_SUPPORTED	Not Support

10.2 RCCopyRawObjPrmToStgMedium

Description This function writes the current development parameter values into the specified storage. The client can set the development parameter value of a RAW object as the one of other RAW objects by calling RCCopyStgMediumPrmToRawObj().

Syntax `cdCAPI RCCopyRawObjPrmToStgMedium (cdHRAW hRAW, cdStgMedium* pTargetStg)`

Parameter

hRAW cdHRAW; A handle for the RAW object.

pTargetStg cdStgMedium*; specify the pointer to cdStgMedium structure.

Returns cdError: cdOK if successful; otherwise, the return value is one of the bellow error list.

Error Type	Comments
cdINVALID_HANDLE	Invalid handle
cdINVALID_PARAMETER	Parameter error

Revision History/Date	Corrections	Reviser	Remarks

cdMEM_ALLOC_FAILED	Memory allocation error
cdNOT_SUPPORTED	Not Support

10.3 RCCopyStgMediumPrmToRawObj

Description This function sets the development parameter value of the Raw object which is retrived by calling RCCopyRawObjPrmToStgMedium() into the the specified RAW object.

Syntax **cdCAPI RCCopyStgMediumPrmToRawObj(cdStgMedium* pStgMedium, cdHRAW hTargetRaw)**

Parameter
pStgMedium cdStgMedium*; specify the pointer to cdStgMedium structure.

hTargetRaw cdHRAW; A handle for the open device.

Returns cdError: cdOK if successful; otherwise, the return value is one of the bellow error list.

Error Type	Comments
cdINVALID_HANDLE	Invalid handle
cdINVALID_PARAMETER	Parameter error
cdMEM_ALLOC_FAILED	Memory allocation error
cdNOT_SUPPORTED	Not Support

10.4 RCWriteParamValueFile

Description This function writes the client specified development parameter values into the specified storage.

Syntax **cdCAPI RCWriteParamValueFile (cdHRAW cdDevT_ParamItemID cdStgMedium* hRAW, PrmID, pDst)**

Parameter
hRAW cdHRAW; A handle for the RAW object.
PrmID cdDevT_ParamItemID; specify the ID of development parameter item.
Only specification of the parameter below the present condition is possible.

Value	Meaning
kPrmID_WBCoeffs	WB Coefficients
kPrmID_4ptRGBToneCurve	4PT RGB Tone Curve
kPrmID_12ptRGBToneCurve	12PT RGB Tone Curve

Revision History/Date	Corrections	Reviser	Remarks

pDst cdStgMedium* ; specify the pointer to cdStgMedium structure.

Returns cdError: cdOK if successful; otherwise, the return value is one of the bellow error list.

Error Type	Comments
cdINVALID_HANDLE	Invalid handle
cdINVALID_PARAMETER	Parameter error
cdMEM_ALLOC_FAILED	Memory allocation error
cdNOT_SUPPORTED	Not Support

10.5 RCReadParamValueFile

Description This function sets the development parameter value of the Raw object which is retrived by calling RCWriteParamValueFile () into the the specified RAW object.

Syntax cdCAPI RCReadParamValueFile (cdStgMedium* pSrc,
cdDevT_ParamItemID PrmID,
cdHRAW hRAW)

Parameter
pSrc cdStgMedium* ; specify the pointer to cdStgMedium structure.

PrmID cdDevT_ParamItemID; specify the ID of development parameter item.
Only specification of the parameter below the present condition is possible.

Value	Meaning
kPrmID_WBCoeffs	WB Coefficients
kPrmID_4ptRGBToneCurve	4PT RGB Tone Curve
kPrmID_12ptRGBToneCurve	12PT RGB Tone Curve

hRAW cdHRAW; A handle for the RAW object.

Returns cdError: cdOK if successful; otherwise, the return value is one of the bellow error list.

Error Type	Comments
cdINVALID_HANDLE	Invalid handle
cdINVALID_PARAMETER	Parameter error
cdMEM_ALLOC_FAILED	Memory allocation error
cdNOT_SUPPORTED	Not Support

Revision History/Date	Corrections	Reviser	Remarks

11. Memory Utility

This section describes the functions to use a memory effectively.

11.1 RCUnFocusRAWObject

Description This function releases only the resource that the specified RAW object allocates. Resources such as intermediate data are allocated to the RAW object by RCDevelop() and some functions. So the client must call this function in order to release the resources.

Syntax `cdCAPI RCUnFocusRAWObject(cdHRAW hRAW)`

Parameter
hRAW cdHRAW; A handle for the RAW object.

Returns cdError: cdOK if successful; otherwise, the return value is one of the bellow error list.

Error Type	Comments
cdINVALID_HANDLE	Invalid handle
cdMEM_ALLOC_FAILED	Memory allocation error
cdNOT_SUPPORTED	Not Support

Revision History/Date	Corrections	Reviser	Remarks